

## ابهامات و اطلاعاتی که لحاظ نشده است:

- 1- مشکلات شبکه ای:  
ممکن است که اینترنت دانشجو کند باشد و این مشکل اصلا از سمت سرور نباشد. که این که باید زیر 3 ثانیه از سمت کلاینت پاسخ را بشنود منطقی نیست. این محدودیت زمانی باید از سمت سرور اعمال و تست شود.
- 2- داده ها در پایگاه داده:  
حجم داده ها در پایگاه داده و نحوه ی توزیع آن ها تاثیر زیادی روی زمان دارد.
- 3- نحوه ی ارسال درخواست ها:  
یکی از مواردی که گفته نشده این است که درخواست ها چگونه ارسال می شوند آیا به صورت هم زمان ارسال می شوند یا در بازه های متفاوتی ارسال می شوند. آیا باید به صورت موازی پردازش شوند؟ آیا لازم است برای درخواست ها صف تشکیل شود؟
- 4- ارتباط با سامانه های دیگر:  
در این فرآیند انتخاب واحد ممکن است لازم باشد با سامانه های دیگر مانند سامانه ی احراز هویت تعامل شود. این که این تعامل چگونه است و آن سامانه ها چقدر طول می کشد که جواب دهند در زمان پاسخ تاثیر می گذارد.
- 5- زیرساخت و سخت افزار:  
یکی از مواردی که روی زمان پاسخ سامانه اثر می گذارد زیرساخت است و این که از چه سخت افزار هایی استفاده می شود.
- 6- زمان اوج فعالیت سامانه:  
این که زمان اوج فعالیت سامانه کی است و در این زمان چه کار های دیگری انجام می شود روی زمان پاسخ دادن سامانه تاثیر دارد.
- 7- ساختار داده ها:  
اطلاعاتی مثل رشته دانشجو، ترم تحصیلی، وضعیت ثبت نام و ظرفیت کلاس ها نیز روی پیچیدگی درخواست ها اثر می گذارند.

- 8- تعداد و نوع درخواست‌های هم‌زمان: وقتی می‌گوییم ۱۰۰۰ دانشجو هم‌زمان در حال انتخاب واحد هستند، مشخص نمی‌شود. آیا همه آن‌ها درخواست‌های ساده‌ای دارند (مثل اضافه کردن یک درس به لیست) یا درخواست‌های پیچیده‌تری (مثل بررسی تعارضات در برنامه هفتگی)؟ آیا برخی دانشجویان درخواست‌های تکراری ارسال می‌کنند؟ مثلاً فشردن مکرر دکمه "انتخاب واحد".
- 9- ترتیب درخواست‌ها: درخواست‌هایی که اول ارسال می‌شوند ممکن است سریع‌تر پردازش شوند، در حالی که درخواست‌های بعدی منتظر صف بمانند.
- 10- بار سامانه (Load): علاوه بر انتخاب واحد، ممکن است سامانه هم‌زمان کارهای دیگری انجام دهد، مثلاً: پردازش اطلاعات اساتید. / نمایش گزارش‌ها به مدیر سیستم. / اجرای فرایندهای خودکار مثل بررسی ظرفیت کلاس‌ها. این وظایف اضافی ممکن است منابع سرور را درگیر کند و زمان پاسخ را افزایش دهد.
- 11- منظور از در فرایند انتخاب واحد بودن چیست؟ آیا این تعداد فقط درخواست‌های فعال هم‌زمان را شامل می‌شود؟ یا ممکن است بخشی از دانشجویان در حال مرور سامانه باشند، در حالی که دیگران به صورت فعال درخواست ارسال می‌کنند؟

### **چرا زمان پاسخ ممکن است متغیر باشد؟**

1. **تفاوت در حجم داده‌ها:** اگر پایگاه داده حجم زیادی از اطلاعات قدیمی را نگهداری کند، جست‌وجو و پردازش کندتر خواهد بود.
2. **تفاوت در نوع درخواست‌ها:** برخی درخواست‌ها مثل انتخاب یک واحد ساده، زمان کمی می‌گیرند، ولی درخواست‌هایی که نیاز به بررسی تعارضات یا ظرفیت کلاس دارند، پیچیده‌تر هستند.
3. **اختلال‌های شبکه:** سرعت اینترنت کاربران و بار شبکه بین کلاینت و سرور می‌تواند زمان پاسخ را تحت تأثیر قرار دهد.

4. **تفاوت در سخت افزار:** سروری که سامانه روی آن اجرا می شود، اگر قدرت پردازشی کافی نداشته باشد یا حافظه کافی برای پردازش همزمان ۱۰۰۰ درخواست نداشته باشد، زمان پاسخ طولانی تر خواهد شد.

### چگونه این مشکل را حل کنیم؟

برای حل این مشکل نیازمند همکاری و تعامل درست بین مشتری و توسعه دهنده است. ابتدا باید نیازمندی ها شفاف شود و ابهام نداشته باشد. بین مشتری و توسعه دهنده باید جلسه ای برگزار شود و در این جلسه جزئیات درخواست مشتری بررسی شود و ابهام ها جواب داده شوند و سعی شود که یک user story از آن خارج شود. این موضوع وظیفه ی هم مشتری و هم توسعه دهنده است و باید با تعامل با هم حل شود. مشتری باید خواسته ی خود و نیازمندی ها را به طور کامل و واضح بیان کند توسعه دهنده هم باید این خواسته را بررسی کند و اگر مواردی گفته نشده بود به مشتری اطلاع دهد و از او بخواهد که بگوید، در واقع وظیفه توسعه دهنده این است که داده های مورد نیازش را از دل مشتری بیرون بکشد و مشتری هم باید در دسترس باشد که این کار امکان پذیر باشد.

### راه حل های دیگر:

1- **تحلیل سناریوها:** مشتری باید اطلاعات دقیق تری از رفتار کاربران و سناریوهای مختلف ارائه دهد. مثلاً: آیا همه دانشجویان در یک زمان مشخص درخواست ارسال می کنند؟ ترکیب درخواست ها چطور است؟ چند درصد درخواست ساده و چند درصد پیچیده هستند؟

2- **تست بارگذاری (Load Testing):** سامانه باید با داده های واقعی شبیه سازی شود تا بتوان زمان پاسخ در شرایط مختلف را ارزیابی کرد.

### 3- بهینه سازی سامانه

- **بهینه‌سازی پایگاه داده:** با استفاده از شاخص‌ها (Indexes) و طراحی مناسب جداول، سرعت جست‌وجوی داده‌ها افزایش می‌یابد.
- **توزیع بار (Load Balancing):** استفاده از چندین سرور به جای یک سرور می‌تواند بار را بین سرورها توزیع کند.
- **کشینگ (Caching):** ذخیره موقت داده‌هایی که بیشتر درخواست می‌شوند (مثل لیست واحدهای ارائه‌شده) می‌تواند زمان پاسخ را کاهش دهد.

#### 4- تعریف معیارهای عملکرد (Performance Metrics)

- معیارهای عملکرد دقیق باید مشخص شوند، مثلاً: زمان پاسخ حداکثر برای ۹۵٪ درخواست‌ها کمتر از ۳ ثانیه باشد. استفاده از منابع سرور از ۸۰٪ تجاوز نکند.

### وظیفه مشتری است یا توسعه‌دهنده؟

مشتری و توسعه‌دهنده باید با هم همکاری کنند تا به نیازمندی‌هایی برسند که هم عملی باشند و هم قابل پیاده‌سازی. اگر هر یک از طرفین وظیفه خود را به درستی انجام ندهند، احتمالاً سامانه عملکرد مناسبی نخواهد داشت.

#### وظیفه مشتری:

- تعریف دقیق نیازمندی‌ها و ارائه اطلاعات دقیق‌تر درباره سناریوها.
- شفاف‌سازی انتظارات عملکردی و محدودیت‌های بودجه و منابع.

#### وظیفه دولوپر:

- تحلیل نیازمندی‌های مشتری و شناسایی اطلاعات گم‌شده.
- اجرای تست‌های مختلف برای شبیه‌سازی سناریوهای واقعی.
- ارائه راه‌حل‌های فنی مثل بهینه‌سازی‌ها و کشینگ.

### توصیف نیازمندی‌های جنبه‌های مختلف

1- Performance Efficiency: سامانه باید بتواند در اوج مصرف به درخواست ها در کمترین زمان ممکن پاسخ دهد. مثلا سامانه باید قادر باشد درخواست های اخذ یک واحد درسی از طرف دانشجویان را در زمانی کمتر از ۳ ثانیه پاسخ دهد، حتی در صورت وجود ۱۰۰۰ درخواست همزمان.

برای درستی سنجیش هم Load Testing انجام شود:

- ۱۰۰۰ درخواست همزمان شبیه سازی شود.
- زمان پاسخ برای هر درخواست ثبت شود.
- اطمینان حاصل شود که ۹۵٪ درخواست ها در کمتر از ۳ ثانیه پاسخ داده می شوند.

روش درستی سنجی: با استفاده از ابزار های تست می توان تعداد بالای کاربران و درخواست ها را شبیه سازی کرد و زمان پاسخ را ارزیابی کرد.

2- Compatibility:

1- سامانه باید بتواند اطلاعات مورد نیاز مانند اطلاعات درس ها و دانشجویان را از سامانه های دیگر مثل بهستان یا ایلرن دریافت کند و موارد لازم را در آن ها بنویسد.

روش درستی سنجی: باید از تست های integration استفاده کرد و انتقال داده ها بین سامانه و سامانه های دیگر بررسی شود که فرمت داده ها درست است و درخواست ها و پاسخ ها بین سامانه ها در زمان درستی انجام می شوند.

برای اندازه گیری آن میتوان حساب کرد که چند درصد اطلاعات از سامانه های دیگر به صورت اتوماتیک و چند درصد آن fail شده و نیاز است به صورت دستی وارد شود.

2- سامانه باید بتواند با سیستم احراز هویت مرکزی دانشگاه هماهنگ شود و دسترسی های غیر مجاز را محدود کند.

روش درستی سنجی: یکبار با استفاده از یوزر احراز هویت نشده تلاش به ورود به سامانه میکنیم و باید دسترسی ما محدود شده باشد و بار دیگر با استفاده

از یوزر احراز هویت شده وارد میشویم و باید با توجه به Authorization ما به ما دسترسی در سامانه داده شود.

3- Interaction Capability: باید کاربران ما (به عنوان مثال مسئول آموزش دانشکده) بتوانند به درستی از قابلیت های برنامه استفاده کنند که در زمان 15 دقیقه آموزش دیده اند (مثلا 15 دقیقه ویدیو آموزشی).

روش درستی سنجی: با دادن زمان و ویدیو آموزشی به یک کاربر جدید که قبلا با این حوزه آشنا بوده (یک مسئول آموزش) باید بتواند کار هایی که ما از او میخواهیم را بتواند انجام دهد.

انجام تست قابلیت یادگیری (Usability Test) با گروهی از دانشجویان جدید. به هر دانشجو یک وظیفه مشخص (مثل انتخاب ۳ واحد درسی) داده شود. زمان لازم برای انجام موفقیت آمیز عملیات ثبت شود. بررسی شود که حداقل ۹۰٪ دانشجویان عملیات را در زمان تعیین شده انجام دهند.

4- Reliability: در صورت قطع سرویس سامانه باید بتواند در زمان کمی وضعیت کاربر را بازیابی کرده و کاربر بتواند کار خود را بدون از دست دادن اطلاعات ادامه دهد و سامانه در کمتر از ۲ دقیقه بازیابی شود.

روش درستی سنجی: می توان در هنگام کار با سامانه سرویس آن را قطع کرد و دوباره وصل کرد و وضعیت سامانه بررسی شود. شبیه سازی یک قطعی ناگهانی در سرور (مثل قطع جریان برق).

پس از راه اندازی مجدد سامانه، بررسی شود که اطلاعات انتخاب واحد دانشجو بدون از دست دادن داده بازیابی شده است. زمان بازیابی نیز با ابزارهای نظارتی (Monitoring Tools) ثبت شود.

5- Security: تمامی کارهای کاربران باید با احراز هویت انجام شود تا لاگ کار های کاربر به نام خود کاربر ثبت شود و مشخص باشد هر فرد چه کار هایی انجام داده است. سامانه باید تمامی تراکنش های مربوط به انتخاب واحد را ثبت کند و اطلاعات

مربوط به هویت دانشجو، زمان انتخاب و جزئیات عملیات به‌گونه‌ای ذخیره شود که امکان انکار عملیات توسط دانشجو یا سامانه وجود نداشته باشد.

روش درستی سنجی: تعدادی کاربر کارهای متفاوتی را با سامانه انجام دهند و فایل لاگ‌ها بررسی شود.

شبیه‌سازی یک سناریوی امنیتی که دانشجو بخواهد انکار کند واحدی انتخاب کرده است و تأیید صحت اطلاعات از لاگ سامانه.

اگر بخواهیم به صورت کمی به قضیه نگاه کنیم میتوانیم یک سری اکشن مهم را لیست کرده و ببینیم چند درصد این اکشن‌ها در لاگ‌های سیستم آمده‌اند و فرد نمیتواند آن را انکار کند و به یک عدد میرسیم.

6- Flexibility: سامانه باید روی دستگاه‌های مختلف بتواند اجرا شود (مثلاً روی وب باشد) همچنین از لحاظ UI بتواند رزولوشن‌های مختلف را هندل کند و تجربه کاربری را خراب نکند. سامانه باید قابلیت نصب و اجرا بر روی زیرساخت‌های مختلف (مانند سیستم‌عامل‌های ویندوز، لینوکس) را داشته باشد.

روش درستی سنجی: با بالا آوردن سامانه روی لپ‌تاپ و موبایل و تبلت و همچنین تست کردن آن در orientation‌های مختلف آنها مطمئن شویم که تمام امکانات چه ظاهری (مثل UI) و چه سمت backend (مثل ذخیره‌سازی یک گزارش در سیستم یا کلیپ‌بورد آن) به درستی قابل اجرا هستند.

نصب سامانه بر روی حداقل 5 سیستم عامل مختلف مثل مک ویندوز لینوکس اندروید و iOS

بررسی صحت عملکرد سامانه در هر محیط با اجرای سناریوهای انتخاب واحد.

مقایسه زمان پاسخ و نرخ موفقیت در هر محیط.